

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. **(Currently Amended)** A computing system, the computing system including:
a processor;
a memory; and
a virtual machine which is in communication with the processor, the virtual machine being arranged to enable two or more jobs, **which are associated with two or more applications**, to run thereon, wherein the virtual machine is arranged to create a heap in the memory for each **one of the two or more jobs** job that run[[s]] on the virtual machine.
2. (Original) A computing system according to claim 1 wherein the virtual machine is scaleable.
3. (Original) A computing system according to claim 1 wherein the virtual machine includes a jobs manager, a class manager, and a heap manager.
4. (Original) A computing system according to claim 3 wherein the heap manager manages substantially all heaps in the memory that are created by the virtual machine.
5. (Original) A computing system according to claim 4 wherein the heap manager is arranged to allow an object allocated on a first heap created by the virtual machine to be visible to a second heap created by the virtual machine.
6. (Original) A computing system according to claim 5 wherein the heap manager uses an object router to exchange data between the first heap and the second heap.

7. (Original) A computing system according to claim 3 wherein the class manager is arranged to enable a class associated with the virtual machine to be shared by the one or more jobs that are arranged to run on the virtual machine.

8. **(Currently Amended)** A computing system according to claim 1 wherein the virtual machine is further arranged to enable information to be exchanged between the one or more jobs that are arranged to **[[be]]** run on the virtual machine.

9. **(Currently Amended)** A computing system according to claim 1 wherein the virtual machine is ~~created using the a~~ Java **compliant virtual machine** programming language.

10. (Original) A computing system according to claim 1 wherein the virtual machine is associated with a system heap, and the virtual machine is further arranged to create a system garbage collector, the system garbage collector being arranged to perform garbage collection on the system heap.

11. (Original) A computing system according to claim 10 wherein the virtual machine is further arranged to create an incremental garbage collector for each heap created in the memory, the incremental garbage collector for each heap being arranged to perform garbage collection on its associated heap.

12. (Original) A computing system according to claim 1 wherein the one or more jobs are arranged to execute substantially concurrently.

13. **(Currently Amended)** A virtual machine arranged to operate in cooperation with a computing system, the virtual machine including:

a first mechanism for creating a first job and a second job **which are respectively associated with a first application and a second application**, the first job and the second job being arranged to **concurrently** run **on** ~~with respect to~~ the virtual machine;

a second mechanism, the second mechanism being arranged to provide the at least one job with at least one class that is arranged to be shared between the first job and the second job; and

a third mechanism, the third mechanism being arranged to exchange information between the first job and the second job.

14. (Original) A virtual machine according to claim 13 wherein the first mechanism is further arranged to create a first heap associated with the first job and a second heap associated with the second job, and the third mechanism is further arranged to increase a size of the first heap and to decrease a size of the second heap.

15. (Original) A virtual machine according to claim 14 further including a first garbage collector and a second garbage collector, wherein the first garbage collector is arranged to perform a garbage collection on the first heap and the second garbage collector is arranged to perform a garbage collection on the second heap.

16. (Original) A virtual machine according to claim 14 wherein the size of the first heap and the size of the second heap may be dynamically altered.

17. (Original) A virtual machine according to claim 13 wherein the second mechanism is further arranged to share the at least one class between the first job and the second job.

18. (Original) A virtual machine according to claim 13 wherein at least one of the first job and the second job includes data which is persisted.

19. **(Currently Amended)** A computer-implemented method for executing a first application substantially concurrently with a second application on a virtual machine, the computer-implemented method comprising:

creating a first job on a virtual machine, the first job being associated with the first application;

creating a second job on the virtual machine, the second job being associated with the second application;

creating a first heap, the first heap being associated with the first job; and

creating a second heap, the second heap being associated with the second job.

20. (Original) A computer-implemented method as recited in claim 19, wherein the first job and the second job share at least one class.

21. **(Currently Amended)** A computer-implemented method as recited in claim 19 wherein the first application and the second application are both executed substantially simultaneously.

22. **(Currently Amended)** A computing system, the computing system including:
at least one processor;
a memory; and
a virtual machine which is in communication with the processor, wherein the virtual machine ~~being~~ is arranged to enable two or more jobs applications to run concurrently thereon.

23. **(New)** A computing system as recited in claim 22, wherein the virtual machine is a Java compliant virtual machine.